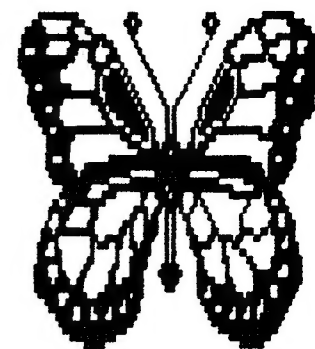
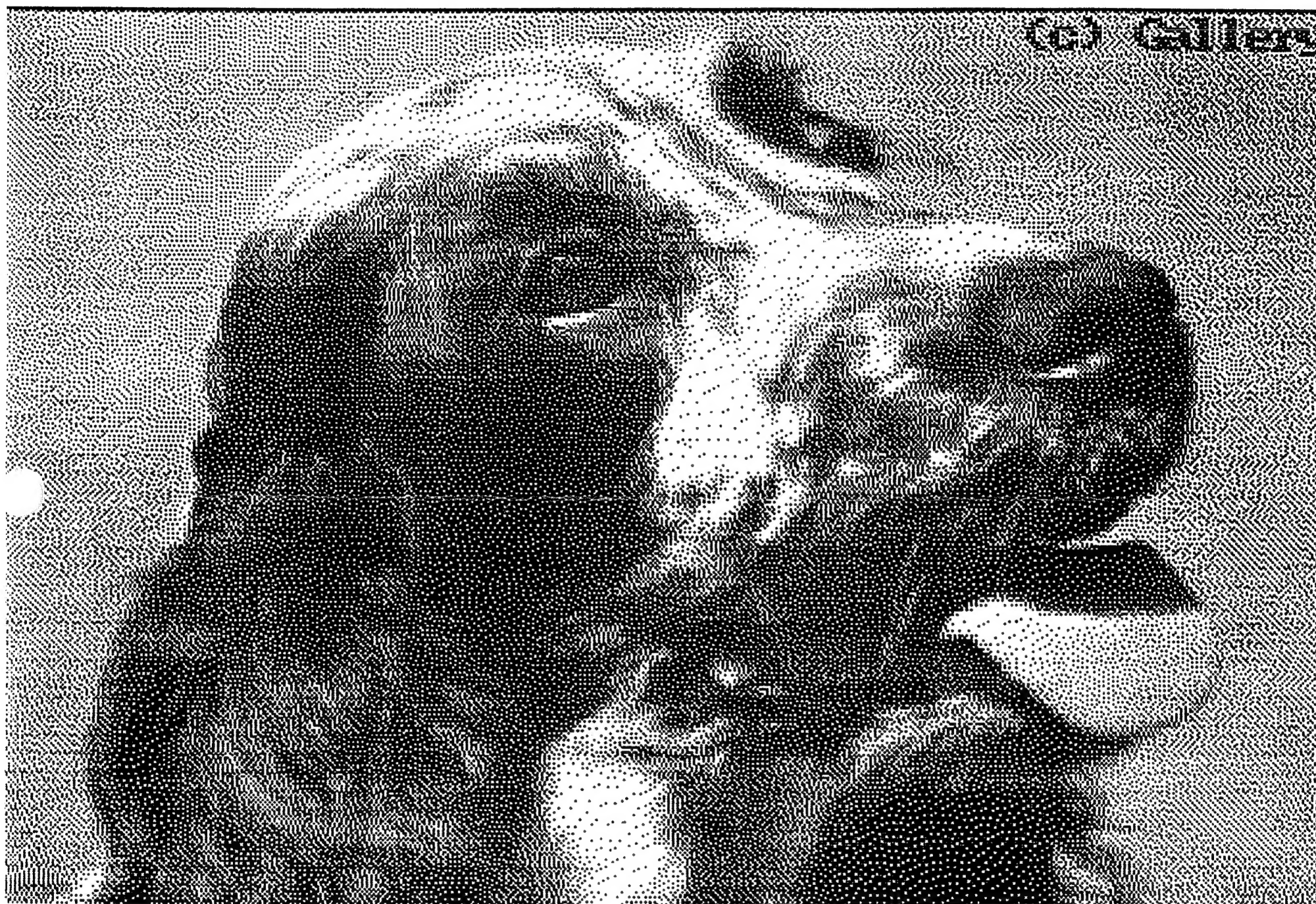


SINC-LINK



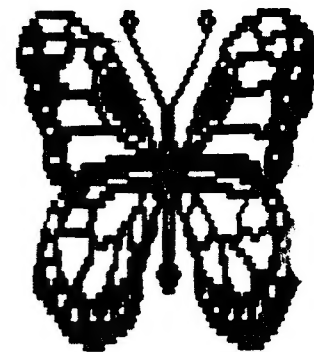
MAY - JUNE '93 VOL 11-3



TORONTO TIMEX-SINCLAIR USERS CLUB



SINC-LINK



MAY - JUNE '93 VOL 11-3

SINC-LINK IS A PUBLICATION OF THE TORONTO TIMEX-SINCLAIR USERS CLUB AND IS ISSUED 6 TIMES A YEAR. CLUB MEMBERS RECEIVE FREE COPIES AS PART OF THE \$20.00 ANNUAL MEMBERSHIP FEE.

NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USER GROUPS.

PLEASE CREDIT THIS PUBLICATION AND THE AUTHOR IF YOU COPY MATERIAL.

THE TS2068 & ZX-81 GROUP MEETS ON THE FIRST WEDNESDAY OF EACH MONTH AT 14 RICHOME COURT, SCARBOROUGH, ONT. 7PM START.

THE QL SIG WILL MEET WEDNESDAY, MAY 18TH AT 586 ONEIDA DRIVE, BURLINGTON, ONT. 7PM START.
JUNE DATE TBA.

SINC-LINK IS PRODUCED ENTIRELY ON SINCLAIR AND TIMEX-SINCLAIR COMPUTERS.

SEND CORRESPONDANCE TO:

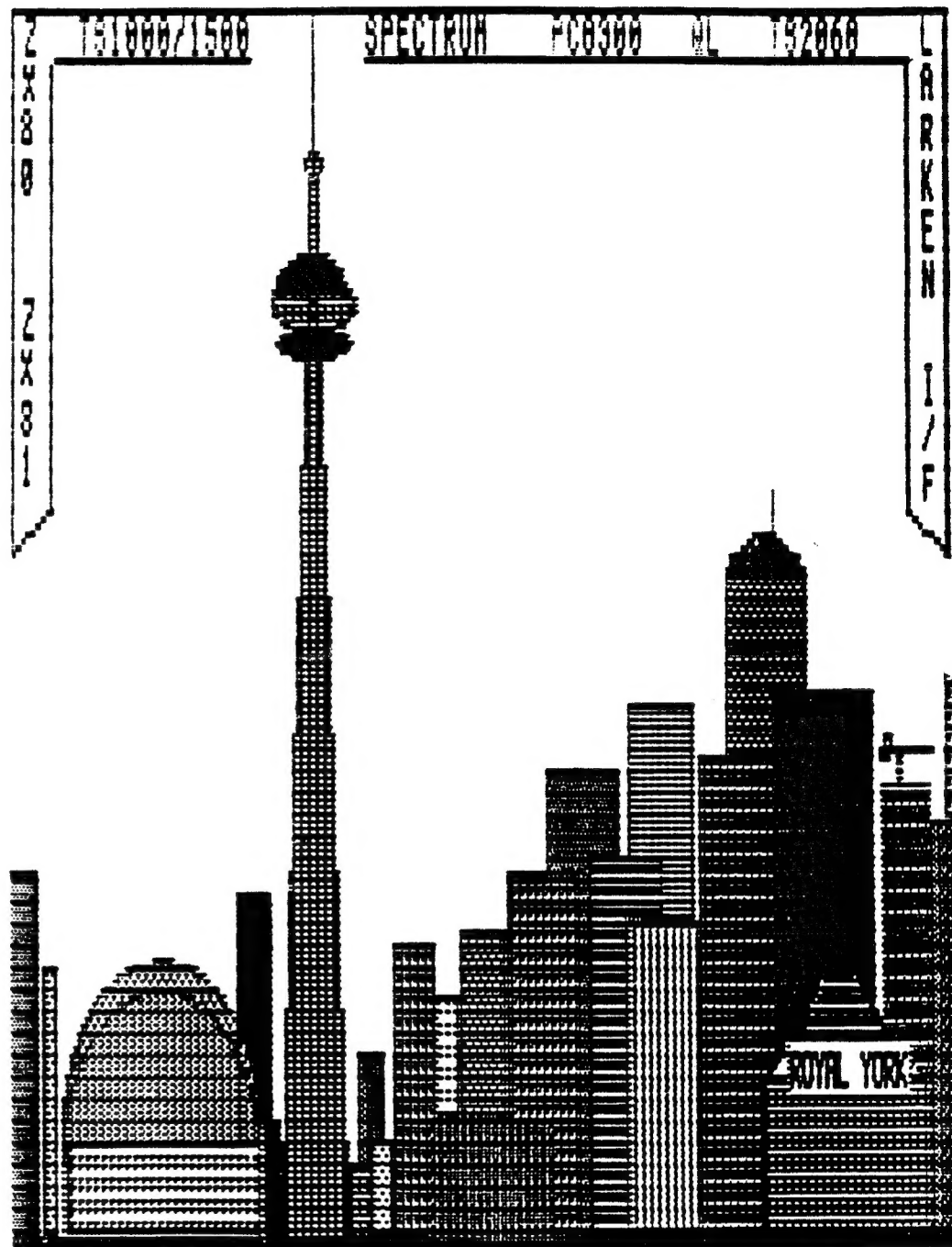
Attention: SINC-LINK
TORONTO TIMEX-SINCLAIR
CLUB, 14 RICHOME
SCARBOROUGH, ONTARIO,
CANADA M1K 2Y1.

EDITOR
USERS
COURT,

EXECUTIVE OFFICERS:

PRESIDENT:
TREASURER:
SECRETARY:
ACTIVITIES:
QL CONTACT:
NEWSLETTER:
LIAISON OFFICER:
(Out-of-town members)

(Area Code 416)
RENE BRUNEAU (531-9749)
BILL LAWSON (444-8772)
GEORGE CHAMBERS (751-7559)
LOU LAFERRIERE (820-3725)
HUGH HOWIE (634-4929)
JEFF TAYLOR (244-8583)
GEORGE CHAMBERS, 14 RICHOME COURT,
SCARBOROUGH, ONTARIO M1K 2Y1
(416- 751-7559)



TORONTO TIMEX-SINCLAIR
USERS CLUB

TORONTO TIMEX-SINCLAIR USERS CLUB

TORONTO TIMEX-SINCLAIR USERS CLUB

SUMMER'S COMING 1993

INDEX VOLUME 11-3

Page 2 - Club Stuff

Page 3 - This Page

Page 4 - Editorial

Page 5 - Larken Disk Index (2068)

Page 6 - Want Ads, Multiplying by 2

Page 7 - QLips - Super Disk Indexer (QL)

Page 8 - Gil Parrish Writes (ZX81 & 2068)

Page 9 - ZX Resources - A to D Converter (ZX81 & 2068)

Page 11 - Did You Know? continued (2068)

Page 15 - TS2068 Programming Tip (2068)

Page 16 - Computer Junkies

Page 17 - Gil Parrish Writes More

Page 19 - Freenet Plugs Ottawa In

Page 21 - ZX Resources - RS232 Revisited (ZX81 & 2068)

Page 23 - Super Disk Labeller (QL)

Page 24 - An American Spectrum

EDITORIAL

Well, the newsletter is definitely late this time. It's partly my fault because I've been so busy at work (overtime ad nauseum). As editor I tend to be a little proprietary about Sinc-Link ("It's MY newsletter and I'll get it out when I'm good and ready!") so offerers of help tend to get their hands bitten. I am mellowing a little, though, and Pres Rene Bruneau will take a crack at producing the July - August issue (I'll be watching him like hawk).

As I said, I am partly to blame for the delay but the other reason is that in the last couple of months we have seen a marked decrease in the number of letters/articles coming in. Why is that? We have more out-of-town members than ever but we're getting less material from them. Surely we can't have discovered everything about our computers. There must be *something* you've found that you haven't seen published, so why not write about it.

Give it a try and I'll try to be more on time, editorially-speaking. 'Nuff said.

RHODE ISLAND UPDATE

As mentioned last issue, here is more info on the QL-fest being held in Newport, Rhode Island.

The get-together will be at the Salvation Army Building on Memorial Boulevard, Saturday June 5th. Show times are 1 PM to 6 PM. Contact Bob Dyl of IQLR for more info (401)-849-3805.

Our own Hugh Howie has reserved a display table for the Toronto Timex-Sinclair Users Club. Hugh will be manning the table, possibly with one or two out-of-town members, handing out lots of promotional stuff about the club and also running an elaborate demonstration of the QL's abilities.

Hugh expects to meet lots of new faces, including some U.K. software and hardware vendors, generate new memberships, attend all the pre- and post-fest talks and dinners and find time to run his demonstrations. Whew, he'll need a week to recover. Watch for his impressions and articles of the fest in the next issue.

BAD NEWS FOR NESQLUG

Just prior to publishing, Hugh phoned to inform me that the Director of the New England Sinclair QL Users Group, Ernest Wider, died suddenly on May 31st. Our condolences to his family and NESQLUG.

SO LONG, SINCLAIR USER

Courtesy of out-of-town member, Robert Shade, I have included an article on the TS2068 lifted from the pages of the last issue (March 1993), of *Sinclair User* magazine from the United Kingdom. This article might explain how a few U.K. vendors (ie Doctor Dark, PRISM PD, etc.) found out about some of the clubs on this side of the pond. Sorry to see another Sinclair-related publication fold. To my knowledge, only *Your Sinclair* remains in publication in the U.K.. Thanks for the info, Robert.

That's all for now ... J.T.

INDEX.B1 By George Chambers
TS2068 Larken

In our TS2068 Larken library, buried in Disks #02 "OMNIBUS" and Disk #33 "PULL-DOWN (Menus) OMNIBUS" is a utility called "index.B1". It is designed to maintain an index of the contents of any collection of Larken disks. One simply feeds the disks into the program, one by one, and it reads and stores the contents of the disks. One can then search, sort, and print out the contents to screen, TS2040 printer, or to a large printer. The program was written by Bob Mitchell some time ago.

A later refined version (V1.11) is found on Disk #47 "UNIQUE PROFRAMS", and the refinements described in this article are applicable to this version. Ask for this disk.

When Bob Mitchell wrote this program he incorporated within it the ability to detect whether the selected printer, TS2040 or large printer, was ON or OFF; and to display the condition on the screen.

The problem with this feature was that not all printers supplied the same signal info as the Fastext 80 used by Bob. And if the signal for the ON condition was not the one Bob had incorporated in his program, one could not continue with the PRINT feature of the program.

There is another feature of the program which may be customised to your particular needs. Many disks contain an AUTOSTART file and/or a "menu" file. An option was provided which allowed these two filenames to be NOT printed out. Later, I expanded the feature to include two more filenames to be similarly excluded from the printout; filenames "taswi", and "help". The program listing below does not provide for changing the AUTOSTART filename, since it is so univerdally used.

The method used to make these changes is to POKE them into the appropriate addresses in the "index.Cc" code. The following listing will do this task. All you need to do is determine your printers' ON/OFF codes and enter these values in the variables. And to enter any changes in the filenames.

To determine the printer ON/OFF signal codes, run this short program:

```
10 PRINT AT 10,10;"        "; AT 10,10; IN 127
20 PAUSE 10: GOTO 10
```

Switch your large printer into it's OFF and ON (ready to print) modes. Write down the values, and then enter these values into the variables LPOFF and LPON in Line 150 of the listing.

Change the IN 127 in the above listing to IN 251. This will check the 2040 printer. Probably this will not need changing, but if it does, then do the same thing. Record the values and place them in variable TSOFF and TSON in Line 160.

```
150 LET LPOFF=127: LET LPON=108
160 LET TSOFF=126: LET TSON=56
170 LET a$="menu": LET b$="help": LET c$="help": LET d$="taswi": REM
Words you may replace
200 POKE 35484,LPOFF: POKE 35570,LPOFF
210 POKE 35516,LPON: POKE 35602,LPON
220 POKE 33426,LPON: POKE 33526,LPON
230 POKE 33394,TSON: POKE 33494,TSOFF
300 LET x=34207: FOR n=1 TO 4: POKE (n+x), CODE a$(n): NEXT n: REM To
replace "menu" with your word.
```


SUPER DISK INDEXER

Recently I came across something we could all do with. A program to help us keep track of our disks, and what is on those disks. This program is called SUPER DISK INDEXER, written by Imre Dominik, and available from Dilwyn Jones Computing at the very modest cost of £12.00 and worth every penny.

The program will keep track of all your disks and cartridges by setting up a database of the contents of your media. You don't have to type very much as the DISK INDEXER reads the contents of your disk or cartridge direct into the database.

You can then use DISK INDEXER to search for a title anywhere in the files.

The first thing to be done is to put your disks in order and give each one a number, and writing that number on the disk itself. Load up DISK INDEXER and follow the simple instructions. Yes it does take time to do all this, but the hardest part is getting your disks in order, and swapping them in the disk drive. The reading is done so quickly you are kept busy.

This program requires at least 256K extra memory, but as most of us now have that, it is really no hardship.

Can handle up to 999 disks with up to 480 files to each disk - now how many files is that? Of course if you run out of space then you can start up another file under a different title. Have you got that many disk/files? How do you keep track of that number without an indexer to help you?

It is time you had this little beauty, I have found it of real value in tracing down exactly where a certain file is.

I would not be without it. Now here is what Dilwyn Jones has to say about it in his catalogue.

DISK INDEXER =====

A really easy to use aid to indexing your floppy disks and cartridges. Sets up a database of the contents of your media (reads the contents directly from the disk or cartridge, you don't have to type it all in) and can handle thousands of filenames on up to 999 media (subject to amount of free memory). You can then search, sort (filename or disk number order) and print the contents of the media. Can't remember which disk that file is on? Search for it using Disk Index - if you know the filename, do an "exact" search, if you don't do a "vague" search. Easy to update the database if the contents of a disk is changed. Works with numbered disks - you simply write the number of the disk on the label and enter it when asked by the program as it reads the directory automatically. Can give you lists of disk contents, lists of unused numbers and so on. This program will really help you to get your disks organised! Super Disk Index needs at least 256k extra memory on a QL. NB can only handle up to about 480 files per disk, so the current version may not be adequate for cataloguing ED disks if you pack them with larger numbers of files! Ideal for cataloguing lower density disks and microdrives. Ideal companion program is our Disk Labeller program, described elsewhere in this catalogue.

DISK INDEXER (on disk or mdv) ... £12.00

Dilwyn Jones Computing. 41 Bro Emrys, Tal-y-bont. Bangor. Gynedd. UK. LL57 3YT

Yep, that's Wales, and he takes plastic.

Gilliam Y. Parrish
Route 1, Box 705
Beggs, OK. 74421
U.S.A.

April 5, 1993

Mr. George F. Chambers, Secretary
Toronto Timex Sinclair Users' Group
14 Richome Court
Scarborough, Ontario M1K 2Y1
Canada

Re: Timex/Sinclair Computers

Dear George,

I mentioned last time that I had made an offer for a 300 baud modem/serial interface from Byte-Back's "Make David an Offer" sale, but hadn't heard anything. Well, the package came in the mail the other day, and believe me, it was worth the wait. I had asked David Leech in my order if I could get copies of both 2068 and 1000 terminal programs (assuming he normally provided such software with the modem), since I had both 2068 and 1000 (1500) units. Mr. Leech did indeed send both 2068 and 1000 tapes, not only for terminal operations but also for utilizing other serial devices like a printer with the serial interface. He replied in the letter accompanying the order that the modem was intended for use with either a 2068 or a 1000 unit, but not both, and had to be hardware configured for the appropriate model. However, since I apparently wanted to use it with both kinds of units, he had added a switch for me so I could change back and forth! I had also mentioned in my order that, when I had gotten a parallel printer cable from him a few months earlier, I had not known to request 1000-specific software to accompany it, and as a result he had sent only the software to have the cable work with my 2068. In his package, Mr. Leech tossed in the 1000 parallel cable software and the instruction sheet! This is clearly service well beyond the call of duty; the T/S world is going to be losing not only a vendor but a true T/S enthusiast when the Mr. Leech's closeout is completed. For people who want to get some of the remaining goodies, his address has changed very recently; it is now: 536 Long Terrace, Leesville, SC. 29070 (USA). I'd strongly urge anyone with a need for a peripheral/interface to get an order in now!

Sincerely,



Gil Parrish

ZX81 RESOURCES ANALOGUE to DIGITAL CONVERTER

Rene Bruneau May 1992

The analogue to digital converter is a simple design based on the ADC0804. It will convert voltages up to 5V into a form that your computer can translate and display for you. In the present configuration, a LM335 is used to convert temperatures from 0 to 100 °C to a voltage that the ADC can sample.

CONSTRUCTION:

Loading the components on the PCB is straightforward. Make sure that C1 and C2 are installed correctly and note that pin 1 of U3 faces pin 1 of U1 and U2. An address block (P2) allows you to map the board to port 191 or 127. The temperature probe is assembled using the LM335 mounted in a BIC pen tube. Seal the ends with epoxy or silicon sealer to protect the connections to the LM335.

CALIBRATION:

To calibrate the ADC, you will need a thermometer, a glass of ice and water, and a glass of freshly boiled water.

For a 2068, use the following procedure:

Enter the following program:

```
10 OUT 191,0
20 PRINT AT 0,0; " "
30 PRINT AT 0,0; IN 191
40 GOTO 10
```

On running the program, place the LM335 in the ice water and after a couple of minutes, adjust RV1 until you read 0 on the screen. This is your minimum temperature setting. Now, place the probe in the hot water along with the thermometer and wait for the readings to stabilize. Adjust RV2 to match the reading on the thermometer. Recheck the probe readings in the ice water and repeat the calibration until you are satisfied.

For the ZX81, you have to install a small machine code routine to provide the IN and OUT commands. Use the same procedure as above to calibrate the ADC.

Load the following:

```
1 REM 12345678901234567890
10 FOR X = 16514 TO 16524
20 INPUT A
30 POKE X, A
40 PRINT X; " "; PEEK X
50 NEXT X
```

At the prompt enter these values:
62, 0, 211, 191, 201, 219, 191, 79, 6, 0, 201

Replace lines 10 - 50 with the following:

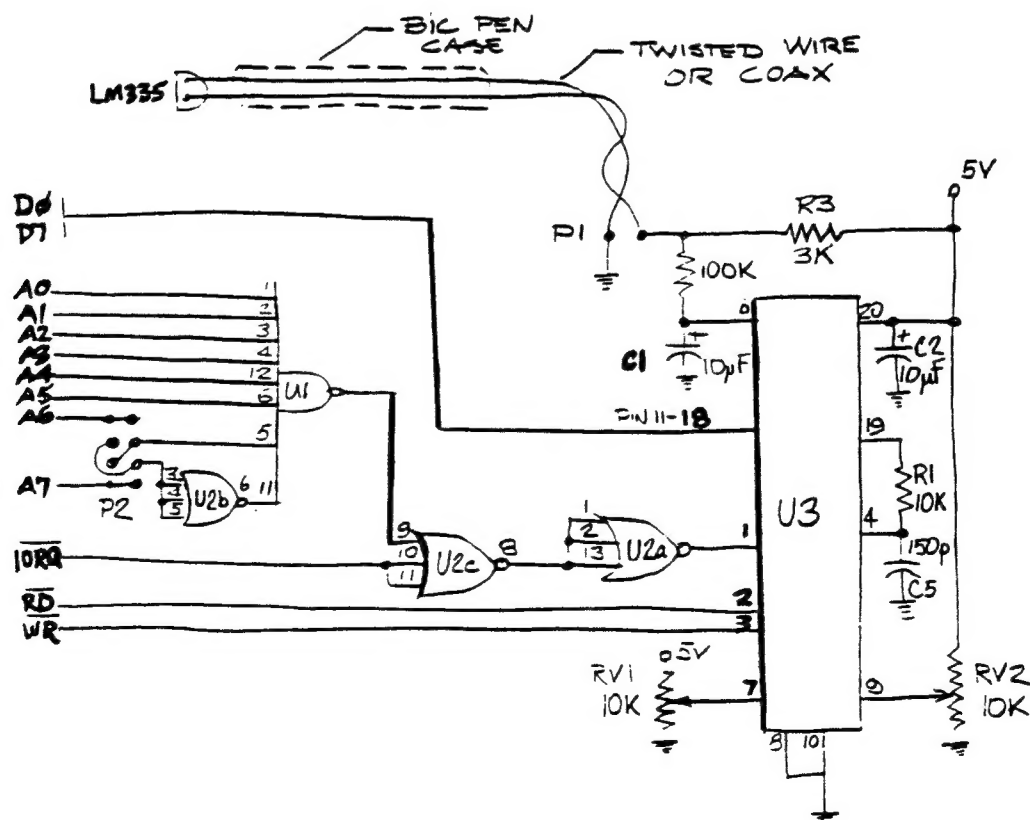
```
10 RAND USR 16514
20 PRINT AT 0,0; " "
30 PRINT AT 0,0; USR 16519
40 GOTO 10
```

Once the ADC is calibrated, you can embellish the program, adding all the bells and whistles you want.

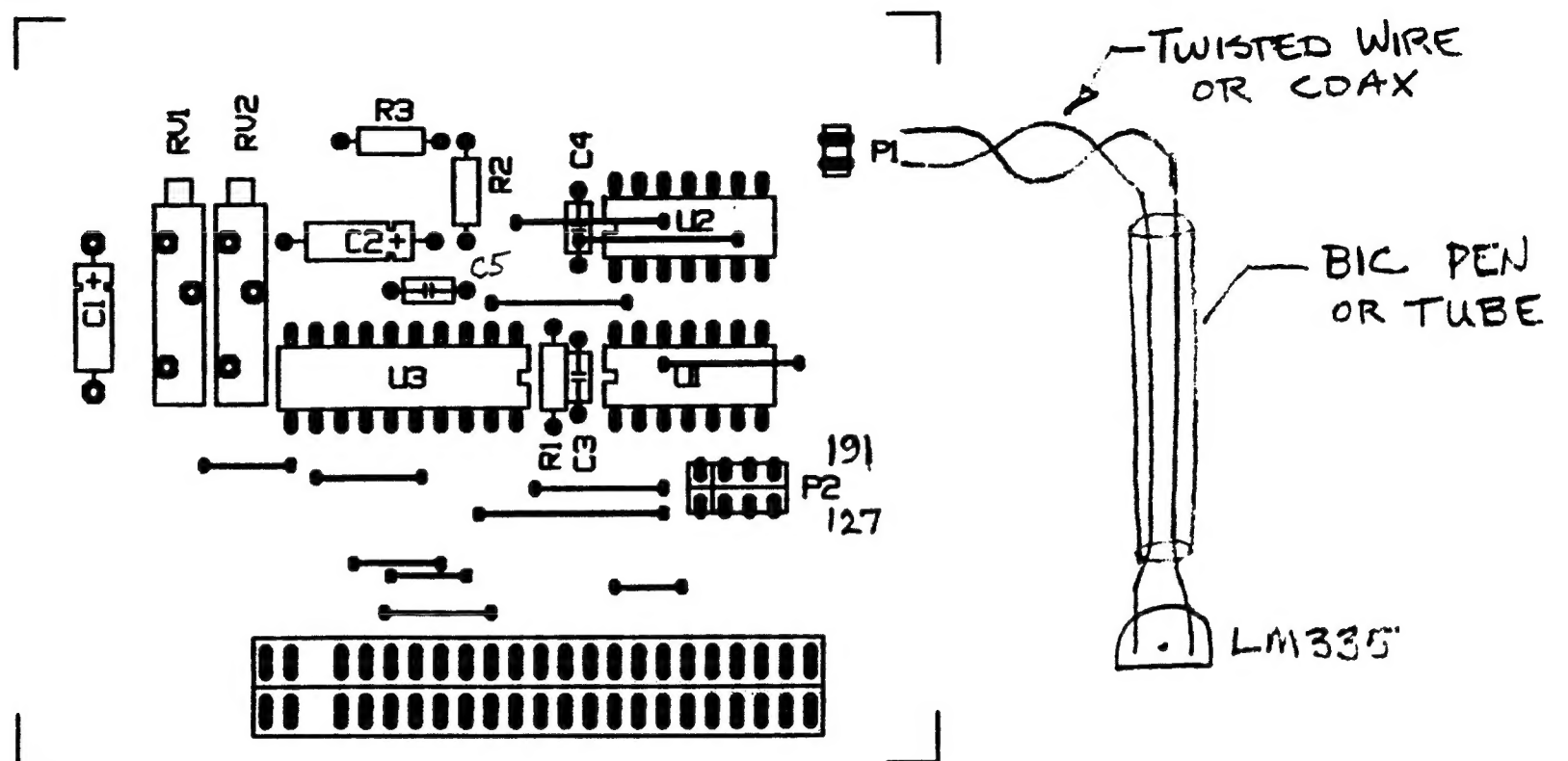
COMPONENT LIST

74LS30	U1	DIL-14
74LS27	U2	DIL-14
ADC0804	U3	DIL-20
LM335	U4	Transistor package
100F	C1	Electrolytic
100F	C2	Electrolytic
100N	C3	Polyester
100N	C4	Polyester
100K	R1	1/4 watt Resistor
680	R2	1/4 watt Resistor
820K	R3	1/4 watt Resistor
100K	RV1	Tripot - 20 turn
100K	RV2	Tripot - 20 turn
SIL-02	P1	Connector for LM335
SIL-04x2	P2	Address block

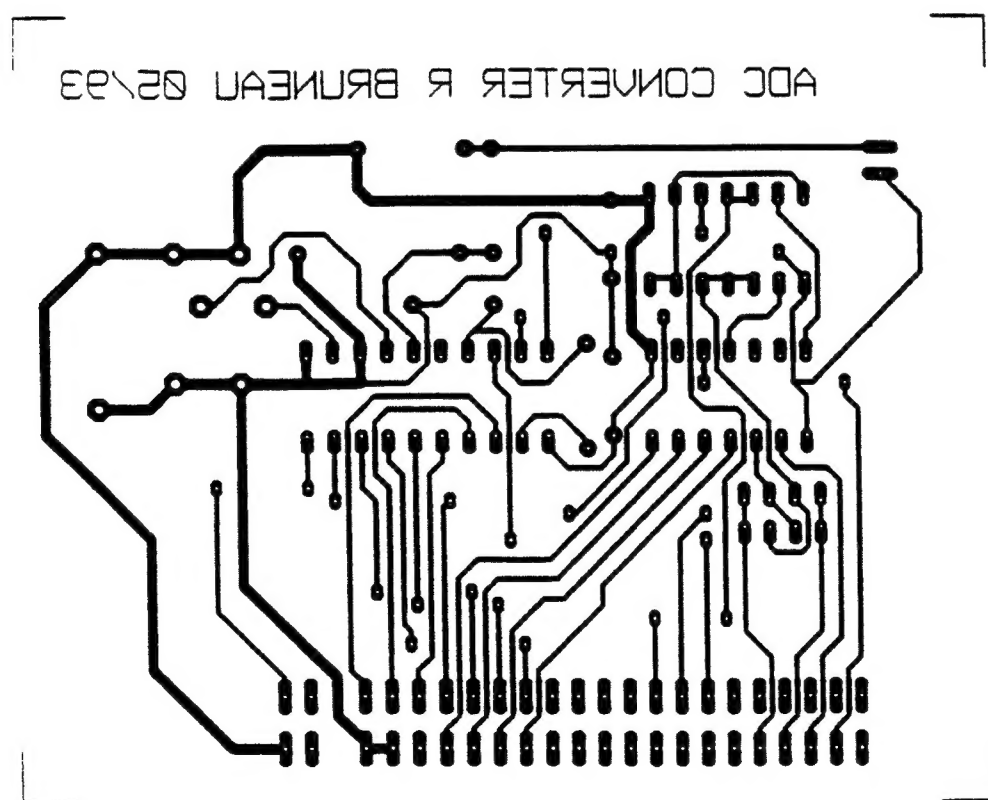
Misc. ZX81 edge connector, coax or twisted wire, jumper wire, BIC pen casing



ANALOG TO DIGITAL CONVERTER



COMPONENT LAYOUT



PCB LAYOUT

ADC CONVERTER R BRUNAU 02723

continued from last issue

name	Dec	Hex	Instr-Dec	Remarks (5)	name	Dec	Hex	Instr-Dec	Remarks
	1469	05BD	LD (8243), HL ;	temp4		1667	0683	JR NC, 1677	
	1472	05C0	CALL 1635			1669	0685	LD HL, (8318) ;	lenth
	1475	05C3	JR 1567			1672	0688	LD (8241), HL ;	temp2
not bas	1477	05C5	CP 67	; C code		1675	068B	JR 1702	
	1479	05C7	JP NZ, 2299	; ERR inval file n		1677	068D	LD HL, (8241) ;	temp2
	1482	05CA	LD HL, (23645);	ch add		1680	0690	LD DE, 5090	
	1485	05CD	LD A, (HL)			1683	0693	OR A	
	1486	05CE	CP 170	; screen\$		1684	0694	SBC HL, DE	
	1488	05D0	JR NZ, 1498			1686	0696	JR C, 1702	
	1490	05D2	LD DE, 16384	; scrn address		1688	0698	LD (8241), HL ;	temp2
	1493	05D5	LD HL, 6912			1691	069B	LD A, 10	
	1496	05D8	JR 1548			1693	069D	LD BC, 5090	
m code	1498	05DA	RST 32	; next char		1696	06A0	LD (8318), BC ;	lenth
	1499	05DB	LK HL, (23645);	chadd		1700	06A4	JR 1709	
	1502	05DE	LD A, (HL)			1702	06A6	LD HL, (8241) ;	temp2
	1503	05DF	CP 58	; :		1705	06A9	LD (8318), HL ;	lenth
	1505	05E1	JR Z, 1541			1708	06AC	XOR A	
	1507	05E3	CP 13			1709	06AD	LD C, A	
	1509	05E5	JR Z, 1541			1710	06AE	LD HL, (8318) ;	lenth
	1511	05E7	CALL 1026	; evalu		1713	06B1	LD A, H	
	1514	05EA	LD (8243), BC ;	temp4		1714	06B2	OR L	
	1518	05EE	LD HL, (23645);	chadd		1715	06B3	RET Z	
	1521	05F1	LD A, (HL)			1716	06B4	LD A, C	
	1522	05F2	CP 44			1717	06B5	LD HL, 8328	
	1524	05F4	JR NZ, 1536			1720	06B8	LD DE, (8243) ;	temp4
	1526	05F6	RST 32	; next char		1724	06BC	LD BC, (8318) ;	lenth
	1527	05F7	CALL 1026	; evalu		1728	06C0	LDIR	
	1530	05FA	LD (8241), BC ;	temp2		1730	06C2	AND A	
	1534	05FE	JR 1555			1731	06C3	RET Z	
	1536	0600	LD HL, (8326) ;	totlen		1732	06C4	LD HL, (8243) ;	temp4
	1539	0603	JR 1552			1735	06C7	LD BC, 5090	
	1541	0605	LD DE, (8316) ;	despin		1738	06CA	OR A	
	1545	0609	LD HL, (8326) ;	totlen		1739	06CB	ADD HL, BC	
	1548	060C	LD (8243), DE ;	temp4		1740	06CC	LD (8243), HL ;	temp4
	1552	0610	LD (8241), HL ;	temp2		1743	06CF	JP 1608	
	1555	0613	LD HL, (8241) ;	temp2	Shrink	1746	06D2	CALL 1063	; 1st byte inchanls
	1558	0616	LD A, H			1749	06D5	CP 244	
	1559	0617	OR L			1751	06D7	JR Z, 1758	; spec
	1560	0618	JR Z, 1596			1753	06D9	LD DE, 5968	; del rec (2068)
	1562	061A	CALL 1635			1756	06DC	JR 1778	
	1565	061D	JR 1596			1758	06DE	LD DE, 6632	; del rec (spectrum)
	1567	061F	LD HL, (8324) ;	var offset		1761	06E1	JR 1778	
	1570	0622	LD DE, (23635);	prog	Grow	1763	06E3	CALL 1063	; 1st byte inchanls
	1574	0626	ADD HL, DE			1766	06E6	CP 244	
	1575	0627	LD (23627), HL;	vars		1768	06E8	JR Z, 1775	; spec
	1578	062A	LD HL, (8321) ;	linenu		1770	06EA	LD DE, 4795	; insert (2068)
	1581	062D	INC HL			1773	06ED	JR 1778	
	1582	062E	LD A, (8322) ;	linenu (hi)		1775	06EF	LD DE, 5717	; make room (spect)
	1585	0631	CP 255			1778	06F2	JP 19	
	1587	0633	JR Z, 1606		decamp	1781	06F5	LD (8240), A	; temp1
	1589	0635	XOR A			1784	06F8	LD HL, 0	
	1590	0636	LD (23618), HL;	newppc		1787	06FB	LD B, 100	; d
	1593	0639	LD (23620), A ;	nsppc		1789	06FD	CALL 1804	
	1596	063C	CALL 1013	; endoln		1792	0700	LD B, 10	
	1599	063F	LD HL, 23623 ;	subppc		1794	0702	CALL 1804	
	1602	0642	INC (HL)			1797	0705	LD A, (8240) ;	temp1
	1603	0643	JP 65	; gtout		1800	0708	ADD A, 48	
	1606	0646	RST 8 ERR 0			1802	070A	RST 16	
	1608	0648	LD HL, (8236) ;	start		1803	070B	RET	
	1611	064B	INC HL			1804	070C	LD C, 47	
	1612	064C	LD (8236), HL ;	start		1806	070E	LD A, (8240) ;	temp1
	1615	064F	LD A, 249			1809	0711	INC C	
	1617	0651	CP (HL)			1810	0712	SUB B	
	1618	0652	RET Z			1811	0713	JR NC, 1809	
	1619	0653	LD A, (HL)			1813	0715	ADD A, B	
	1620	0654	LD (8249), A			1814	0716	SRL (HL)	
	1623	0657	CALL 3807	; nextrk		1816	0718	LD (8240), A ;	temp1
	1626	065A	LD A, (8249)			1819	071B	LD A, C	
	1629	065D	LD HL, 8221 ;	curtrk		1820	071C	RST 16	
	1632	0660	CP (HL)			1821	071D	RET	
	1633	0661	JR NZ, 1623		Doserr	1822	071E	LD A, (8194) ;	nmi flag
LDdata	1635	0663	CALL 1084	; ldbuf		1825	0721	CP 1	
	1638	0666	LD A, (8224) ;	errnu		1827	0723	JR NZ, 1832	
	1641	0669	CP 25			1829	0725	LD SP, 23550	
	1643	066B	JP Z, 2320			1832	0728	PUSH HL	
	1646	066E	LD HL, (8318) ;	lenth		1833	0729	CALL 1013	; endoln
	1649	0671	LD DE, 5090			1836	072C	XOR A	
	1652	0674	OR A			1837	072D	CALL 3268	; lprint
	1653	0675	SBC HL, DE			1840	0730	LD HL, (23651);	stkbot
	1655	0677	JR NC, 1677			1843	0733	LD (23653), HL;	stkend
	1657	0679	LD HL, (8318) ;	lenth		1846	0736	LD HL, (23698);	membot
	1660	067C	LD DE, (8241) ;	temp2		1849	0739	LD (23656), HL;	mem
	1664	0680	OR A			1852	073C	POP HL	
	1665	0681	SBC HL, DE			1853	073D	CALL 1922	; print to

name	Dec	Hex	Instr-Dec	Remarks (6)	name	Dec	Hex	Instr-Dec	Remarks
	1856	0740	LD HL, 23611	; flags		2120	0846	LD A, (8224)	; errnu
	1859	0743	RES 5, (HL)	; key hit		2123	084B	CP 10	
	1861	0745	LD HL, 0			2125	084D	JR NZ, 2133	
	1864	0748	LD (IY+38), H	; x ptr (hi)		2127	084F	LD HL, 1974	; "T File Not Found"
	1867	074B	LD (IY+55), H	; flagx		2130	0852	JP 1822	; doserr
	1870	074E	LD (23563), HL	; defad		2133	0855	LD HL, (8241)	; temp2
	1873	0751	INC HL			2136	0858	PUSH HL	
	1874	0752	LD (23574), HL	; strms+6		2137	0859	CALL 2711	; zer0
	1877	0755	LD HL, 4927			2140	085C	POP HL	
	1880	0758	LD DE, 5808			2141	085D	INC HL	
	1883	075B	CALL 1046	; roms		2142	085E	LD (HL), 254	
	1886	075E	LD HL, 23665	; flagx		2144	0860	LD HL, (8245)	; temp6
	1889	0761	RES 5, (HL)	; line of prog		2147	0863	INC HL	
	1891	0763	LD HL, 23612	; tvflag		2148	0864	LD (HL), 249	
	1894	0766	SET 5, (HL)	; clhs		2150	0866	CALL 3651	; savebf
	1896	0768	LD SP, (23613)	; errsp		2153	0869	JP 1596	
	1900	076C	POP HL		GTfil	2156	086C	LD HL, 8226	; progm
	1901	076D	LD HL, 3865	; 2068		2159	086F	LD A, 32	
	1904	0770	LD DE, 4937	; spectrum		2161	0871	LD B, 10	
	1907	0773	CALL 1063	; 1st byte inchnls		2163	0873	LD (HL), A	
	1910	0776	CP 244			2164	0874	INC HL	
	1912	0778	JR Z, 1918	; spectrum		2165	0875	DJNZ 2163	
	1914	077A	PUSH HL			2167	0877	RST 32	; next char
	1915	077B	JP 65	; gtout		2168	0878	LD HL, 7151	; 2068 !
	1918	077E	PUSH DE			2171	087B	LD DE, 7308	; spec > temio
	1919	077F	JP 65	; gtout		2174	087E	CALL 1046	; roms
PRINTTO	1922	0782	LD A, 13			2177	0881	LD HL, (23653)	; stkend
	1924	0784	RST 16			2180	0884	DEC HL	
	1925	0785	LD A, (HL)			2181	0885	LD B, (HL)	
	1926	0786	CP 46	; .		2182	0886	DEC HL	
	1928	0788	RET Z			2183	0887	LD C, (HL)	
	1929	0789	RST 16			2184	0888	DEC HL	
	1930	078A	INC HL			2185	0889	LD D, (HL)	
	1931	078B	JR 1925			2186	088A	DEC HL	
	1933		"S Protect Error"	<pointer @ 2494		2187	088B	LD E, (HL)	
	1951		"CRC ER"	<pointer @		2188	088C	DEC HL	
	1959		"U Disk Full"	<pointer @ 2424		2189	088D	LD (23653), HL	; stkend
	1974		"T File Not Found"	<pointer @ 2127		2192	0890	EX DE, HL	
	1993		"V Wrong File Type"	<pointer @ 2301		2193	0891	LD DE, 8226	; progm
	2013		"X Cat Data Error"	<pointer @ 2320		2196	0894	PUSH BC	
Vserch	2031	07EF	LD HL, (23627)	; vars		2197	0895	LD A, B	
	2034	07F2	LD A, 128			2198	0896	OR C	
	2036	07F4	CP (HL)			2199	0897	JR Z, 2203	
	2037	07F5	JR Z, 2096			2201	0899	LDIR	
	2039	07F7	LD A, (8247)			2203	089B	POP BC	
	2042	07FA	CP (HL)			2204	089C	LD A, 34	; "
	2043	07FB	RET Z			2206	089E	LD (DE), A	
	2044	07FC	LD A, (HL)			2207	089F	RET	
	2045	07FD	AND 224		cmdck	2208	08A0	CALL 2156	; gtfil
	2047	07FF	CP 224			2211	08A3	LD HL, 8226	; progm
	2049	0801	JR Z, 2071			2214	08A6	LD B, 7	
	2051	0803	CP 192			2216	08A8	LD A, (HL)	
	2053	0805	JR Z, 2077			2217	08A9	CP 94	
	2055	0807	CP 160			2219	08AB	JR Z, 2299	
	2057	0809	JR Z, 2083			2221	08AD	CP 46	
	2059	080B	CP 128			2223	08AF	JR Z, 2230	
	2061	080D	JR Z, 2077			2225	08B1	INC HL	
	2063	080F	CP 96			2226	08B2	DJNZ 2216	
	2065	0811	JR Z, 2088			0228	08B4	JR 2299	
	2067	0813	CP 64			2230	0886	INC HL	
	2069	0815	JR Z, 2094			2231	0887	LD A, (HL)	
	2071	0817	LD DE, 19			2232	08B8	PUSH HL	
	2074	081A	ADD HL, DE			2233	08B9	LD B, A	
	2075	081B	JP 2034			2234	08BA	LD HL, (23645)	; chadd
	2077	081D	CALL 2102			2237	08BD	CP 65	; A
	2080	0820	INC HL			2239	08BF	JR NZ, 2248	
	2081	0821	JR 2034			2241	08C1	LD A, (HL)	
	2083	0823	INC HL			2242	08C2	CP 228	; data
	2084	0824	BIT 7, (HL)			2244	08C4	JR NZ, 2301	
	2086	0826	JR Z, 2083			2246	08C6	JR 2280	
	2088	0828	LD DE, 6			2248	08C8	CP 67	; C
	2091	082B	ADD HL, DE			2250	08CA	JR NZ, 2263	
	2092	082C	JP 2034			2252	08CC	LD A, (HL)	
	2094	082E	JR 2077			2253	08CD	CP 175	; code
	2096	0830	LD A, 128			2255	08CF	JR Z, 2280	
	2098	0832	LD (8248), A			2257	08D1	CP 170	; screen\$
	2101	0835	RET			2259	08D3	JR Z, 2280	
	2102	0836	INC HL			2261	08D5	JR 2301	
	2103	0837	LD DE, (HL)			2263	08D7	CP 66	; B
	2104	0838	INC HL			2265	08D9	JR NZ, 2299	
	2105	0839	LD D, (HL)			2267	08DB	LD A, (HL)	
	2106	083A	ADD HL, DE			2268	08DC	CP 202	; line
	2107	083B	RET			2270	08DE	JR Z, 2280	
ERASE	2108	083C	CALL 2158	; gtfil		2272	08E0	CP 58	; :
	2111	083F	LD A, 32			2274	08E2	JR Z, 2280	
	2113	0841	LD (DE), A			2276	08E4	CP 13	
	2114	0842	CALL 2444	; wprot		2278	08E6	JR NZ, 2301	
	2117	0845	CALL 2307	; indir		2280	08C8	POP HL	

name	Dec	Hex	Instr-Dec	Remarks (7)	name	Dec	Hex	Instr-Dec	Remarks
	2281	08E9	INC HL		WPROt	2444	098C	LD A, (8195)	; dvsel
	2282	08EA	LD A, (HL)			2447	098F	CP 128	
	2283	08EB	CP 94			2449	0991	JR NZ, 2481	
	2285	08ED	JR Z, 2299			2451	0993	LD A, 71	
	2287	08EF	LD (23681), A			2453	0995	OUT 7, A	
	2290	08F2	INC HL			2455	0997	IN A, 244	
	2291	08F3	LD A, 34			2457	0999	LD C, A	
	2293	08F5	CP (HL)			2458	099A	LD A, 240	
	2294	08F6	JR NZ, 2299			2460	099C	OUT 244, A	
	2296	08F8	LD (HL), 32			2462	099E	LD HL, 32770	
	2298	08FA	RET			2465	09A1	LD (HL), A	
	2299	08FB	RST 8 ERR 15	; invalid file name		2466	09A2	LD B, (HL)	
	2301	08FD	LD HL, 1993	; "V Wrong File Type		2467	09A3	LD (HL), 0	
	2304	0900	JP 1822	; doserr		2469	09A5	LD A, C	
indir	2307	0903	CALL 1212	; drv0		2470	09A6	OUT 244, A	
	2310	0906	CALL 1084	; ldbuf		2472	09A8	XOR A	
	2313	0909	LD A, (8224)	; errnu		2473	09A9	OUT 7, A	
	2315	090C	CP 25			2475	09AB	LD A, 240	
	2318	090E	JR NZ, 2326			2477	09AD	CP B	
Fatal	2320	0910	LD HL, 2013	; "X Cat Data Error"		2478	09AE	JR NZ, 2494	
	2323	0913	JP 1822	; doserr		2480	09C0	RET	
	2326	0916	LD A, (8195)	; dvsel		2481	09B1	CALL 1212	; drv0
	2329	0919	CP 128			2484	09B4	IN A, 16	
	2331	091B	JR Z, 2343			2486	09B6	RRA	
	2333	091D	LD A, (8326)	; totlen		2487	09B7	JR C, 2481	
	2336	0920	RRA			2489	09B9	IN A, 16	
	2337	0921	RRA		?	2491	09BB	BIT 6, A	
	2338	0922	AND 3			2493	09BD	RET Z	
	2340	0924	LD (8202), A	; aspd		2494	09BE	LD HL, 1933	; "S Protect Error
	2343	0927	LD HL, 8328			2497	09C1	JP 1822	; doserr
	2346	092A	INC HL		SAVE	2500	09C4	CALL 2208	; cmdck
	2347	092B	LD A, 255		Ssubr	2503	09C7	CALL 2444	; wprot
	2349	092D	CP (HL)			2506	09C7	CALL 2307	; indir
	2350	092E	JR Z, 2359			2509	09CD	LD A, (8224)	; errnu
	2352	0930	LD A, 250			2512	09D0	CP 10	
	2354	0932	CP (HL)			2514	09D2	CALL Z, 2404	; newet
	2355	0933	JR Z, 2386			2517	09D5	CALL 2711	; zero
	2357	0935	JR 2346			2520	09D8	CALL 3651	; savebf
	2359	0937	LD (8241), HL	; temp2		2523	09DB	LD A, (8194)	; nmi flag
	2362	093A	INC HL			2526	09DE	CP 11	
	2363	093B	LD DE, 8226	; progm		2528	09E0	RET Z	
	2366	093E	LD B, 9			2529	09E1	LD HL, 8226	; progm
	2368	0940	LD A, 254			2532	09E4	INC HL	
	2370	0942	CP (HL)			2533	09E5	LD A, 46	
	2371	0943	JR Z, 2346			2535	09E7	CP (HL)	
	2373	0945	LD A, (DE)			2536	09E8	JR NZ, 2532	
	2374	0946	CP (HL)			2538	09EA	INC HL	
	2375	0947	JR NZ, 2346			2539	09EB	LD A, (HL)	
	2377	0949	INC DE			2540	09EC	CP 65	; A
	2378	094A	INC HL			2542	09EE	JR Z, 2598	; save an array
	2379	094B	DEC B			2544	09F0	CP 66	; B
	2380	0940	JR NZ, 2373			2546	09F2	JP Z, 2792	; save basic program
	2382	094E	LD (8245), HL	; temp6		2549	09F5	LD HL, (23645)	; chadd
	2385	0951	RET			2552	09F8	LD A, (HL)	
	2386	0952	LD A, 10			2553	09F9	CP 170	; screen\$
	2388	0954	LD (8224), A	; errnu		2555	09FB	JR Z, 2584	; save a screen
	2392	0957	RET			2557	09FD	CP 175	; code
movdr	2393	0958	LD HL, (8241)	; temp2		2559	09FF	JP NZ, 2299	; err 15 invalid f n
	2395	095B	LD DE, 8250	; directory		2562	0A02	RST 32	; next char
	2398	095E	LD BC, 34			2563	0A03	CALL 1026	; evalu
	2401	0961	LDIR			2566	0A06	LD (8243), BC	; temp4
	2402	0963	RET			2570	0A0A	RST 32	; next char
newet	2404	0984	XOR A			2571	0A0B	CALL 1026	; evalu
	2405	0965	LD (8224), A	; errnu		2574	0A0E	LD (8241), BC	; temp2
	2408	0968	LD HL, 8328		SMEM	2578	0A12	CALL 2842	
	2411	0968	INC HL			2581	0A15	JP 2958	
	2412	096C	LD A, 254			2584	0A18	LD HL, 16384	
	2414	0965	CP (HL)			2587	0A18	LD (8243), HL	; temp4
	2415	096F	JR Z, 2430			2590	0A1E	LD HL 6912	
	2417	0971	LD A, 250			2593	0A21	LD (8241), HL	; temp2
	2419	0973	CP (HL)			2596	0A24	JR 2578	
	2420	0974	JR Z, 2424			2598	0A26	CALL 2623	
	2422	0976	JR 2411			2601	0A29	LD A, (8248)	
	2424	0978	LD HL, 1959	; "U Disk Full"		2604	0A2C	BIT 7, A	
	2427	097B	JP 1822	; doserr		2606	0A2E	JR NZ, 2621	
	2430	097E	LD DE, 8226	; progm		2608	0A30	LD HL, (8241)	; temp2
	2433	0991	LD B, 9			2611	0A33	INC HL	
	2435	0983	LD A, (DE)			2612	0A34	LD (8241), HL	; temp2
	2436	0934	LD (HL), A			2615	0A37	CALL 2842	
	2437	0985	INC HL			2618	0A3A	JP 2958	
	2438	0986	INC DE			2621	0A3D	RST 8 ERR 2	; variable not found
	2439	0987	DJNZ 2435			2623	0A3F	LD HL, (23645)	; chadd
	2441	0989	JP 2326			2626	0A42	LD A, (HL)	
						2627	0A43	CP 288	; data

name	Dec	Hex	Instr-Dec	Remarks (8)	name	Dec	Hex	Instr-Dec	Remarks
	2629	0A45	JP NZ, 2299	; err invalid file name	2798	0AEE	JRNZ, 2811		
	2632	0A48	RST 32	; next cha	2800	0AF0	RST 32	; next char	
	2633	0A49	LD HL, (23645);	chadd	2801	0AF1	CALL 1026	; evalu	
	2636	0A4C	LD A, (HL)		2804	0AF4	DEC BC		
	2637	0A4D	INC HL		2805	0AF5	LD (23662), HL;	oldppc	
	2638	0A4E	LD A, (HL)		2809	0AF9	JP 2817		
	2639	0A4F	CP 36	; \$	2811	0AFB	LD HL, (23621);	ppc	
	2641	0A51	JR Z, 2667		2814	0AFE	LD (23662), HL;	oldppc	
	2643	0A53	LD A, (23681)		2817	0B01	LD DE, (23635);	prog	
	2646	0A56	CP 36	; \$	2821	0b05	LD HL, (23641);	eline	
	2648	0A58	JP Z, 2301		2824	0B08	PUSH DE		
	2651	0A5B	LD HL, (23645);	chadd	2825	0B09	OR A		
	2654	0A5E	LD A, (HL)		2826	0B0A	SBC HL, DE		
	2655	0A5F	SET 7, A		2828	0B0C	LD (8241), HL ;	temp2	
	2657	0A61	AND 159		2831	0B0F	POP HL		
	2659	0A63	LD (8247), A		2832	0B10	LD (8243), HL ;	temp4	
	2662	0A66	LD (23662), A ;	oldppc	2835	0B13	CALL 2842		
	2665	0A69	JR 2688		2838	0B16	JP 2958		
	2667	0A6B	LD HL, 23681		2841	0B19	RET		
	2670	0A6E	CP (HL)		2842	0B1A	XOR A		
	2671	0A6F	JP NZ, 2301		2843	0B1B	LD (8247), A		
	2674	0A72	LD HL, (23645);	chadd	2846	0B1E	LD HL, (8241) ;	temp2	
	2677	0A75	LD A, (HL)		2849	0B21	LD (8196), HL		
	2678	0A76	OR 192		2852	0B24	LD A, H		
	2680	0A78	RES 5, A		2853	0B25	OR L		
	2682	0A7A	LD (8247), A		2854	0B26	JR NZ, 2860		
	2685	0A7D	LD (23662), A;	oldppc	2856	0B28	POP BC		
	2688	0A80	CALL 2031	; vserch	2857	0B29	JP 1596		
	2691	0A83	LD A, (8248)		2860	0B2C	LD DE, 5090		
	2694	0A86	BIT 7, A		2863	0B2F	LD HL, (8241) ;	temp2	
	2696	0A86	RET NZ		2866	0B32	OR A		
	2697	0A89	LD (8243), HL ;	temp4	2867	0B33	SBC HL, DE		
	2700	0A8C	INC HL		2869	0B35	JR C, 2914		
	2701	0A8D	LD E, (HL)		2871	0E37	LD HL, (8243) ;	temp4	
	2702	0A8E	INC HL		2874	0B3A	LD DE, 8328		
	2703	0A8F	LD D, (HL)		2877	0B3D	LD BC, 5090		
	2704	0A90	INC DE		2880	0B40	LDIR		
	2705	0A91	INC DE		2882	0B42	LD HL, (8243) ;	temp4	
	2706	0A92	LD (8241), DE ;	temp2	2885	0B45	LD (8236), HL ;	start	
	2710	0A96	RET		2888	0B48	LD DE, 5090		
zero0	2711	0A97	CALL 2392	; movdr	2891	0B4B	ADD HL, DE		
	2714	0A9A	LD HL, (8241) ;	temp2	2892	0B4C	LD (8243), HL ;	temp4	
	2717	0A9D	LD BC, 11		2895	0B4F	LD HL, 5090		
	2720	0AA1	ADD HL, BC		2898	0B52	LD (8238), HL		
	2721	0AA1	LD (HL), 249		2901	0B55	LD DE, 5090		
	2723	0AA3	LD HL, 8250 ;	directory	2904	0B58	LD HL, (8241) ;	temp2	
	2726	0AA6	INC HL		2907	0B5B	OR A		
	2727	0AA7	LD A, 253		2908	0B5C	SBC HL, DE		
	2729	0AA9	CP (HL)		2910	0B5E	LD (8241), HL ;	temp2	
	2730	0AAA	JR NZ, 2726		2913	0B61	RET		
	2732	0AAC	INC HL		2914	0B62	LD HL, (8241) ;	temp2	
	2733	0AAD	LD A, 249		2917	0B65	LD A, H		
	2735	0AAF	CP (HL)		2918	0B66	OR L		
	2736	0AB0	JR Z, 2756		2919	0B67	JR NZ, 2928		
	2738	0AB2	LD (8241), HL ;	temp2	2921	0B69	POP HL		
	2741	0AB5	LD C, (HL)		2922	0B6A	LD HL, (8245) ;	temp6	
	2742	0AB6	LD B, 0		2925	0B6D	JP 3114		
	2744	0AB8	LD HL, 8328		2928	0B70	LD HL, (8243) ;	temp4	
	2747	0ABB	ADD HL, BC		2931	0B73	LD DE, 8328		
	2748	0ABC	LD (HL), C		2934	0B76	LD BC, (8241) ;	temp2	
	2749	0ABD	LD HL, (8241) ;	temp2	2938	0B7A	LDIR		
	2752	0AC0	LD (HL), 0		2940	0B7C	LD HL, (8243) ;	temp4	
	2754	0AC3	JR 2732		2943	0B7F	LD (8236), HL ;	start	
	2756	0AC4	LD HL, 8250 ;	directory	2946	0B82	LD HL, (8241) ;	temp2	
	2759	0AC7	INC HL		2949	0B85	LD (8238), HL		
	2760	0AC8	LD A, 253		2952	0B88	LD A, 100 ;	d	
	2762	0ACA	CP (HL)		2954	0B8A	LD (8247), A		
	2763	0ACB	JR NZ, 2759		2957	0B8D	RET		
	2765	0ACD	INC HL		2958	0B8E	LD HL, 8261 ;	directory	
	2766	0ACE	PUSH HL		2961	0B91	LD A, (HL)		
	2767	0ACF	POP DE		2962	0B92	LD (8249), A		
	2768	0AD0	LD B, 21		2965	0B95	LD (8245), HL ;	temp6	
	2770	0AD2	LD HL, 8328		2968	0B98	LD A, 249		
	2773	0AD5	INC HL		2970	0B9A	CP (HL)		
	2774	0AD6	LD A, (HL)		2971	0B9B	JP Z, 2924		
	2775	0AD7	CP 255		2974	0B9E	LD A, (HL)		
	2777	0AD9	JR Z, 2788		2975	0B9F	LD (8221), A ;	curtrk	
	2779	0ADB	CP 245		2978	0BA2	CALL 3571 ;	trac	
	2781	0ADD	JR Z, 2773		2981	0BA5	CALL 1135 ;	encdbf	
	2783	0ADF	LD A, (HL)		2984	0BA8	CALL 3651 ;	savebf	
	2784	0AE0	LD (DE), A		2987	0BAB	CALL 1084 ;	ldbuf	
	2785	0AE1	INC DE		2990	0BAE	LD A, (8234) ;	errnu	
	2786	0AE2	DJNZ 2773		2993	0BB1	AND A		
	2788	0AE4	LD A, 249		2994	0BB2	JR Z, 3064		
	2790	0AE6	LD (DE), A		2996	0BB4	LD A, 3		
	2791	0AE7	RET		2998	0BE6	LD (8223), A ;	attempts counter	
	2792	0AE8	LD HL, (23645);	chadd	3001	0BB9	CALL 1121 ;	clrbf	
	2795	0AEB	LD A, (HL)		3004	0BBC	CALL 3571 ;	trac	
	2796	0AEC	CP 202 ;	line	3007	0BBF	LD BC, (8238)		

TS 2068 PROGRAMMING TIP
by George Chambers

The system variables stored at addresses 23552 to 23755 are used by the computer to keep track of its operation. Look in the TS2068 User Manual, "Appendix D, System Variables" for a list and short description. These stored variables offer quite a few programming possibilities. This article will describe an application for two of them.

The variables being discussed are labelled SPOSN, and are located at addresses 23688 and 23689. Together these two variables store the current PRINT position; that is, where the next character on the screen will be printed. The first address contains the column number, while the second address contains the line number. Well, not the screen positions as we understand them. We would get the screen positions by subtracting the PEEKed value in address 23688 from 24 (for column number), and in address 23689 from 32 (for line number).

In my programming I did a Larken CATalogue, followed by a flashing message 'Press a key for menu'. I wished to remove the message without clearing the rest of the screen.

I could, of course, have simply had the program 'PRINT AT x,y;"' to wipe out the flashing message. But in this application the message could appear on most any line; depending on the size of the CATalogue display. I made use of the variables mentioned above.

Figure 1. is a listing of a short program to demonstrate the possibilities. Let's discuss this listing:

LINE 5: We are assigning a random number to each of variables a and b. The numbers produced by the RND function will be a value always less than 1.

LINE 10. The variables a and b are being used to print "george" at random locations on the screen. The 'a*22' selects a random screen line from 0 to 22. The 'b*26' similarly selects a screen column. (We have used 'b*26', rather than 'b*32', so that the "george" will be printed on one line, i.e. it won't be wrapped around onto the second line. If this happened it would require more complex programming to remove the flashing 'george', and tend to obscure the model.

Note also the semi-colon at the end of 'george'. Without this semi-colon the print position stored in the variable would be at the start of the next line, and the 'covered' would always appear at the beginning of the next line. Try it and see.

LINE 15 Pauses for a moment so you can see the 'george' is present.

LINE 20 Overlays the flashing 'george' with the 'covered' message. Because of the way the system stores the information we have to subtract the PEEKed values from the screen size values of 24 and 32.

Note the '-6' in the line. This is because what we have in the system variable is the present PRINT position. But we want to start our overlaid word 'covered' six positions back, to the start of 'george'.

LINE 30 Self explanatory.

```
5 LET a=RND: LET b=RND
10 PRINT AT INT (a*22),INT (b*26);"george";
15 PAUSE 30
20 PRINT AT 24-(PEEK 23689),32-(PEEK 23688)-6;"covered"
30 PAUSE 30: CLS : GO TO 1
```

E s p e c i a l l y f o r J u n k i e s

by Hugh Howie

(Hey Ma! Some-one has been talking about me - I am in the headlines!)

Are you a Junky?

Now don't get me wrong, there are all kinds of junkies going the rounds.

There is the old fellow who used to go round the neighborhood pushing his little hand cart in front of with his cry of "Rags - Bottles - Bones" Or does your memory not go that far back? That guy was called a Junky. He collected junk. Some times he would be called a Scrappy.

Then there is that other junkie, the one on the drug scene, but we are not interested in that so we will let that fly stick to the wall.

Now that there is another thing. Do you know where that saying came from? That was another oldie, from the time before all those fancy bug killers in a bottle where you press a button and the stuff comes out with a phssst. It comes from the time when we used the old fashioned fly-swat; that was a flat piece of something no matter what that was attached to a stick about a foot or fifteen inches long. When you smacked a fly on the wall with that - splat! - the fly stuck there! (the fly swat was also great for juvenile discipline)

So you just let it stick there. After all, can you think of a better way to keep score of the 'bag' for the day, than to count the "splats" on the wall?

To-day we have different kinds of junkies, ourselves for example, we are Sinclair junkies. We collect Sinclair stuff till it is coming out our ears, (watch it bub) and every time Old papa Sinc comes out with something new we all rush out and grab it. ('cept the byke)

I am going to ask you to have a look at your collection of program you have stored away in all those shoe boxes and drawers. Take 'em out. Now I don't care whether it is Cartridges or Tapes or Disks; treat them all the same. And this is where things can become very interesting, apart from puzzling.

Clear all that other "junk" off that same table, and divide that table into three equal sections. If you want to be sure you have made the division properly, and want to keep things looking neat and tidy and not looking "junky", then draw a pencil mark at the divisions. And label the sections:- Lots. Sometimes. Never.

Now comes the interesting part.

We take all them programs and place them according to your 'labels'. Those programs you use a lot. Those programs you use sometimes. Those programs you never use.

How are you doing? Are the piles of programs increasing evenly? Don't get dis-heartened, keep at it, you are learning. One thing you are learning is that there is a whole stack of stuff you NEVER use - and just take a look at that pile- WOW!

Now look at the pile marked "LOTS" Not too many what? Half a dozen? a few more perhaps? Are there more than dozen. Probably not.

The middle pile is a funny one; some of the stuff in that pile could be in the OFTEN pile, but a lot of it could also be in the NEVER pile.

NOW - Who is the "junky"? You are. I am. We all is! That's right, did you ever think you could be called a "Program Junky"?

Why do we spend so much time and \$\$\$\$ gathering all that stuff (junk)? and never use it.

Puzzle:- -- What to do with that junk --

I have the cure for you, and if you follow my suggestion you will no longer be a "program junky" and you will get rid of all that "junk" gathering dust where-ever it is stored.

Send it to me.

Bye !

930126

Gilliam Y. Parrish
Route 1, Box 705
Beggs, OK. 74421
U.S.A.

May 10, 1993

Mr. George F. Chambers, Secretary
Toronto Timex Sinclair Users' Group
14 Richome Court
Scarborough, Ontario M1K 2Y1
Canada

Re: Timex/Sinclair Computers

Dear George,

I think I found a solution to my (possible) problem of having too small a power supply for that ZX Printer. The manual says the printer needs 9V, 1.2 amps; unfortunately, even the "large" TS/ZX standard power supply is just 1 amp. My discovery came in a second hand shop, in the form of the Mattel Aquarius home computer. You may never have heard of them; they were discontinued almost as soon as they hit store shelves ("One of the shortest lifespans of any computer," according to COMPUTE! magazine). They were a low-end model, with 4K, "chicklet" keyboard, sparse BASIC, designed for cassette storage, etc. They utilized a one-key entry of BASIC commands (actually, CONTROL + the needed key) just like the TS/ZX, but they also allowed typing in the commands letter by letter, a feature I appreciate. One unusual aspect of the model was the printer port; the computer was designed for yet another incarnation of the simple and cheap Alphacom narrow-paper thermal printer (like the T/S 2040), but the connector is a single mini-phono jack. How can a printer get all the signals it needs through a single line?? Anyway, the computer had a cartridge port designed to accept a couple of expansion boxes Mattel had waiting in the wings (to add joysticks, extra RAM, improved sound capability, and on the deluxe expander, disk drives), and for that reason Mattel may have given the model a larger power supply than it really needed. The power supply, which does not disattach (and hence can't get away from the unit in garage sales) states that it produces 8.8VDC, 1.2 amps (as well as minor amounts of other obscure voltages, being .15 amps of 16VDC, and .01 amps of -19VDC). A quick meter check reveals that amperage from the Aquarius unit is indeed significantly higher (and even voltage is a tad HIGHER, not lower) compared to a standard 9V, 1 amp supply. So, it's an easy solution to lack of a 1.2 amp power supply to get an Aquarius, cut off the power supply and cord, and install the right connector for TS/ZX use; the proper wires to solder to the needed mini-phono plug are clearly indicated on the unit (red to center, black to the outside). It's also a cheap solution: since these computers were

never popular, they might be \$2-\$3 if you happen to run across one, and since nobody is actually using these things, you can likely spot one up for grabs if you know to look for them. I picked up two; I'm keeping an Aquarius intact for no particularly good reason, and I've almost completed surgery on the other.

On a different topic, remember the "Hobbit"? That was the Spectrum clone being produced in the former Soviet Union, that was supposed to be "available for export" someday. Well, the day has apparently come. I noticed in the April, 1993 edition of Update! that the Hobbit is now available in the UK through ADB Russian Services, 17 Stortford Rd., Great Dunmow, Essex CM6 1DA, Great Britain. (I assume they must have touched all the legal bases with AMSTRAD before they could sell it there.) It is supposed to be fully 48K Spectrum compatible, but it has 64K RAM, 64K ROM, a QWERTY keyboard but with both Roman and Cyrillic characters on it, built-in CP/M and FORTH as well as BASIC, and the capability to handle either 5 1/4" or 3 1/2" drives as well as a cassette. (I doubt the extra RAM will get much workout; as I reported last year following my trip to the UK, the vast majority of Spectrum software remains 48K compatible despite the sale of many Spectrum units with 128K by AMSTRAD over the years.) The Hobbit is £79.00 and the optional 5 1/4" drive is £59.00; that translates to something like US \$123.25 (CN \$156.50) for the computer and US \$92.00 (CN \$117.00) for the drive. The good old T/S 2068, which can be picked up used for far less money and then equipped with a Spectrum ROM, remains a much lower cost alternative for North Americans who wish to try Spectrum computing-- and isn't it nice to have a computer that's so much in demand that it's still in production in 1993, rather than one that was discontinued in 1984?

By the way, we had discussed whether I wanted to make my new term of membership retroactive to when my last term ended in '92, or to make it effective for the coming year. I asked to have it started as of "now" (around the beginning of April) and not retroactively, so I would be paid up for the coming 12 months. And I think that's what you understood me to say, since you (correctly) did not send past newsletters. However, whoever filled out my membership card (your Treasurer? I can't make out the name), which you forwarded to me last time, apparently did not understand since the card lists my term as 5/92 to 5/93 instead of 4/93 to 4/94. Please don't let my membership expire on me again!!

Sincerely,



Gil Parrish

FREENET PLUGS OTTAWA IN

Starting today, anyone with a computer and modem
can connect with a network of databases - free
by Dominique Lacasse - Citizen business writer

Turn on, boot up, and dial out. "Community computing" is here.

As of today, Ottawa becomes one of only about a dozen cities anywhere in the world to have a non-profit, community computing network. A phone call to a single Ottawa number will plug anyone with a computer and a modem into a network of data-bases and information providers from around the region, across the U.S., and ultimately, the world. Virtually unlimited two-way information - free.

The National Capital FreeNet was to make it's public debut today with a ceremony at regional government headquarters this morning.

It may not be an "electronic highway", but Ottawa's FreeNet is a crucial fork in the road to the Electronic Village, an interactive public library for the computer generation.

Ottawa's network is modelled on successful experiments in community computing in the U.S. It was spearheaded by a group of Carleton University professors and the school's director of computing and information services and is run entirely by volunteers.

After more than a year of planning, fundraising, and scrounging, Ottawa's FreeNet is now a reality, with a powerful host computer and 20 telephone lines to handle the expected rush. There are an estimated 100,000 PCs in households in the greater Ottawa region, and likely just as many in business and government offices.

The network has already been running experimentally at Carleton for several months and has met with wide acceptance. Ottawa's FreeNet has about 1,000 registered users now and about 100 new applications are received every day, according to David Sutherland, National Capital FreeNet chairman and director of computing services at Carleton.

"This thing has gained critical mass", says Sutherland, who only recently decided to double the number of phone lines in time for today's launch as donations and inquiries poured in. "It's had an overwhelmingly positive public response."

Sutherland says he expects registrations to mushroom following today's invitation to the public to join the network.

He says he expects the FreeNet will have at least 10,000 registered users by the end of it's first year. In Cleveland, Ohio, where the first FreeNet was launched in the mid-80's, there are more than 35,000 registered FreeNet users and tens of thousands more who occasionally use the system as "guests".

"We keep hearing we're in the information age", says Sutherland. "These (networks) are the windows that will allow it to come into peoples lives."

Ottawa's FreeNet, which runs on equipment donated by Sun Microsystems and Nepean-based Gandalf Data, already has a mix of databases, from regional government agendas and tender documents and community group bulletins, to exotic items such as news reports from Radio Free Europe and an "artificial intelligence" program on epilepsy from the Federal Communications Research Centre.

And only recently, the (Ottawa) Citizen joined a growing number of information providers experimenting with the system. The Citizen's FreeNet offerings include brief guides to local sports, entertainment and community events, as well as capsule reviews of movies, restaurants, and wines.

The Ottawa Public Library's catalogue is also available online through the FreeNet and the library has installed three terminals at its main branch that will allow people without computers at home or work to explore FreeNet. Discussions are also underway to put terminals in schools throughout the region.

Ottawa's FreeNet also provides a gateway to other FreeNets, including Cleveland's and a new one in Victoria, the only other one in Canada.

Using Internet, a network that links researchers around the world, FreeNet users can visit distant computer systems without incurring long distance charges.

Being free has it's price, however. Because the network only has 20 telephone lines, users may find logging onto the system frustrating, even impossible, at peak times. Sutherland says new lines will be added if demand warrants and sufficient funds can be found.

Some government funding has been received and more is expected, says Sutherland. The goal is to eventually have one full-time paid administrator to manage the network, as several U.S. FreeNets have done.

Sutherland says he hopes the FreeNet will "make the community work better" by promoting the free exchange of information and ideas. He says he hopes the network will make the use of information technology "part of the county's culture, something people are comfortable with for the rest of their lives".

Logging on to FreeNet

You need a computer and a modem to gain access to FreeNet. Use these settings to connect:

Phone (613) 780 3733

Modem Settings:

- Baud Rate: 2400, 1200, or 300
- Data Bits: 8
- Parity: None
- Stop Bits: 1

Important notice: Once connection has been established, you may have to hit the Return or Enter key up to FIVE times.

Log on as "guest" the first time. You can become a registered user by following the instructions available on-line under the "Administration" menu.

This material was retyped from an article in the Ottawa Citizen.

One of our members, Jacques Croteau, writes, "I include an article in the Citizen of Ottawa, about FreeNet. It seems wonderful, but I know only one person who has succeeded in logging on...at 3 o'clock in the morning, before it was advertised!! It seems very busy.

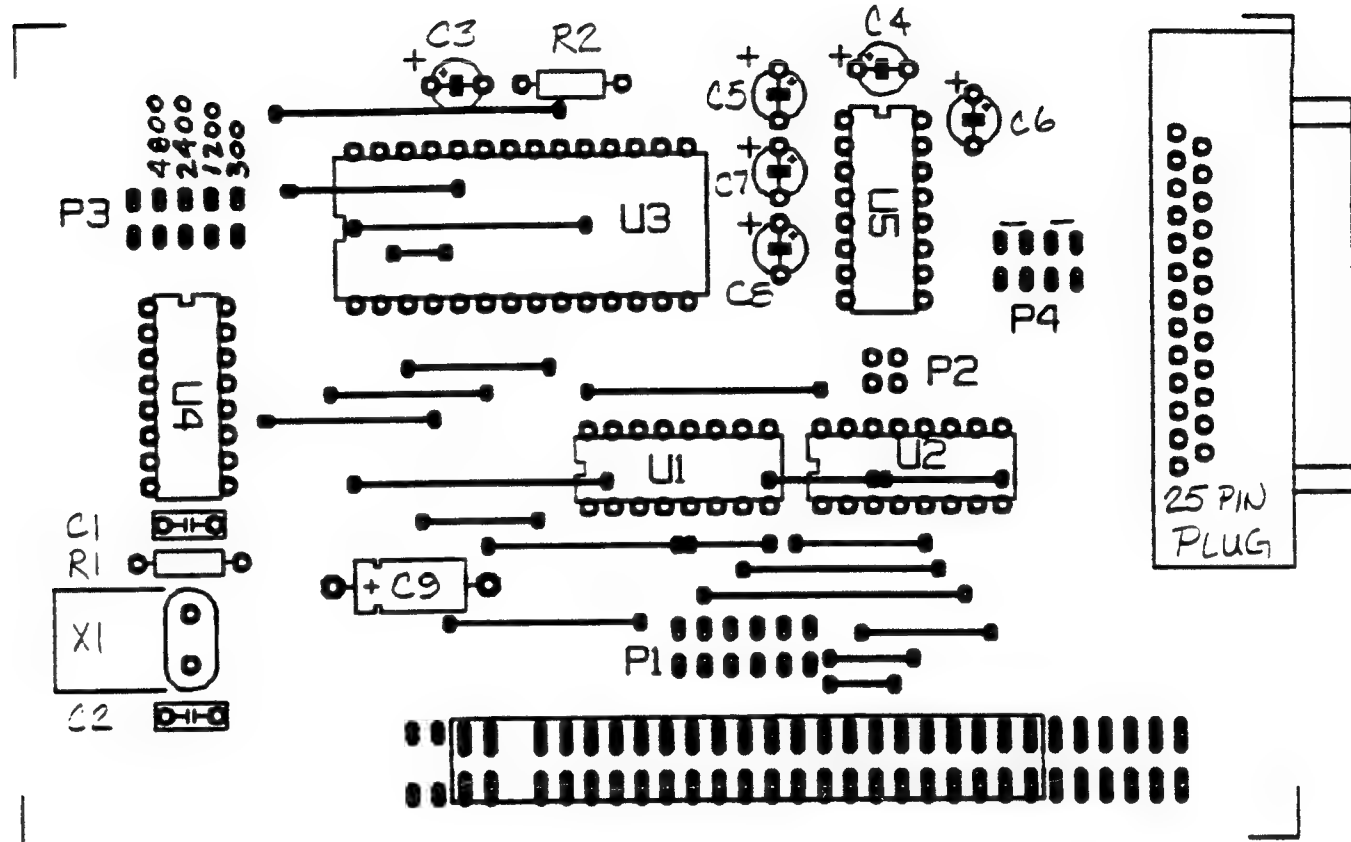
He also penned a note as follows: "New network in Ottawa. I hope you have more success than me in reaching it. I have called about 50 times without result."

"(613) 788 3947 - This number was given on radio for the network....bu always busy."

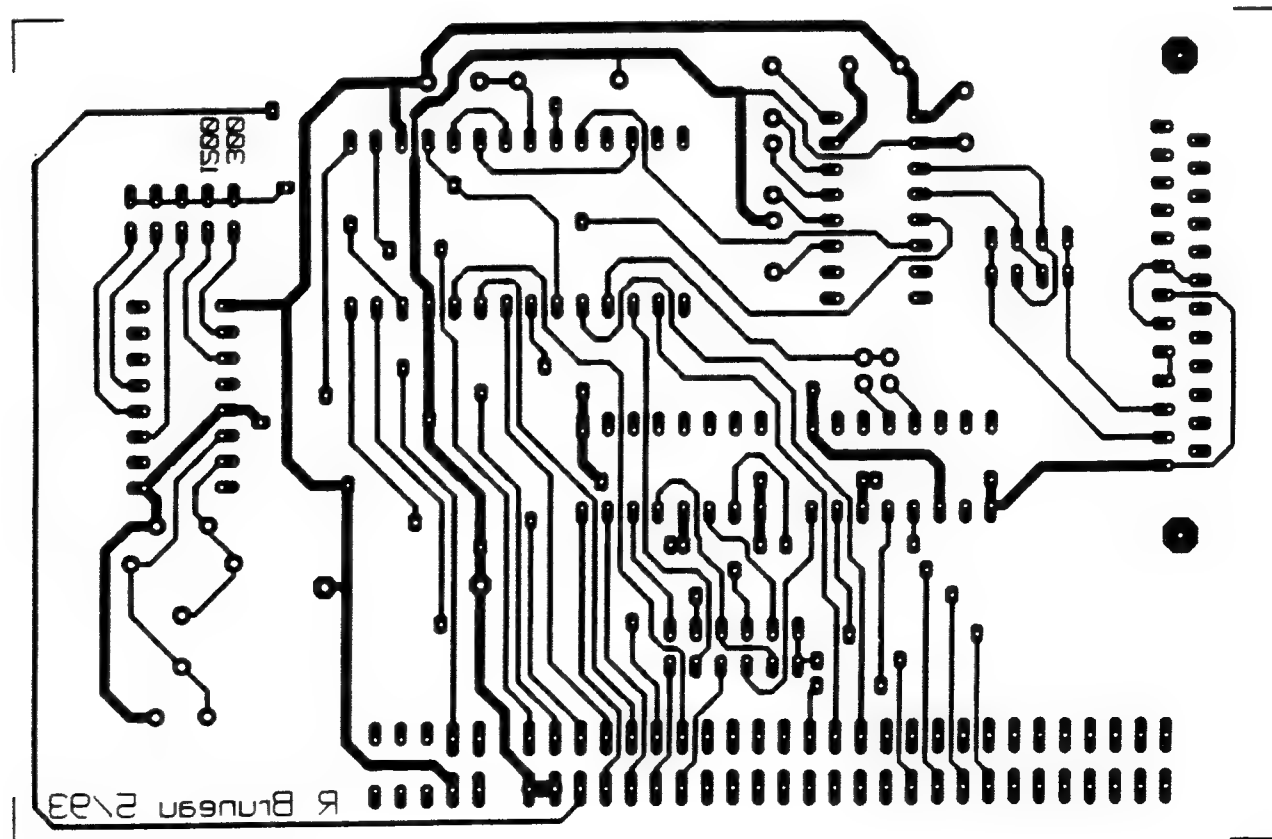
RS232 I/O Revised

Rene Bruneau

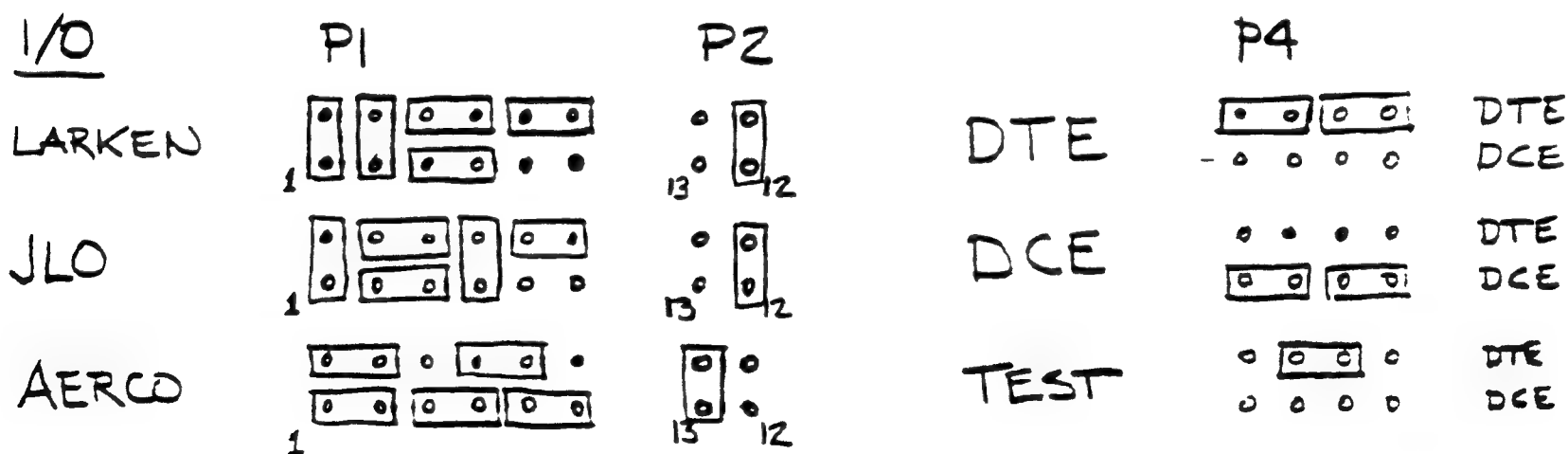
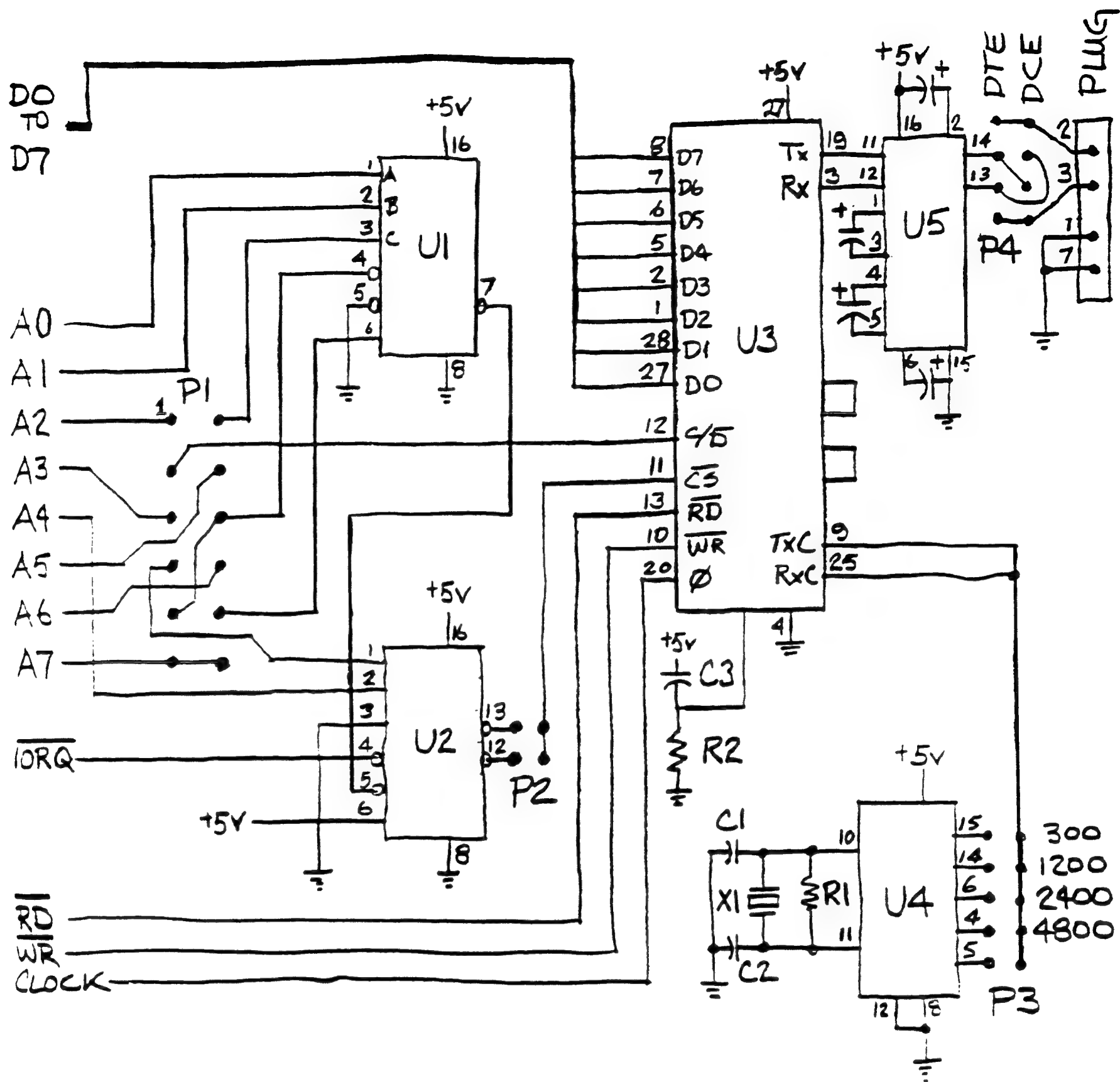
A couple of errors crept into the schematic for the RS232 article found in the last issue of Sinc-Link. The revised circuit diagram is shown below. I also ran into difficulties sourcing the 9.8403 MHz crystal and found that a 4.920 MHz crystal would work, but the PCB had to be revised to get the proper baud rates again.



COMPONENT LAYOUT



PCB LAYOUT



RS-232 CIRCUIT DIAGRAM

SUPER DISK LABELLER

Now this little thing I do not have, but I include it here just in case someone wants to get it at the same time as SUPER DISK INDEXER, and as it is mentioned in my QLIPS message in this issue.

SUPER DISK LABELLER

=====

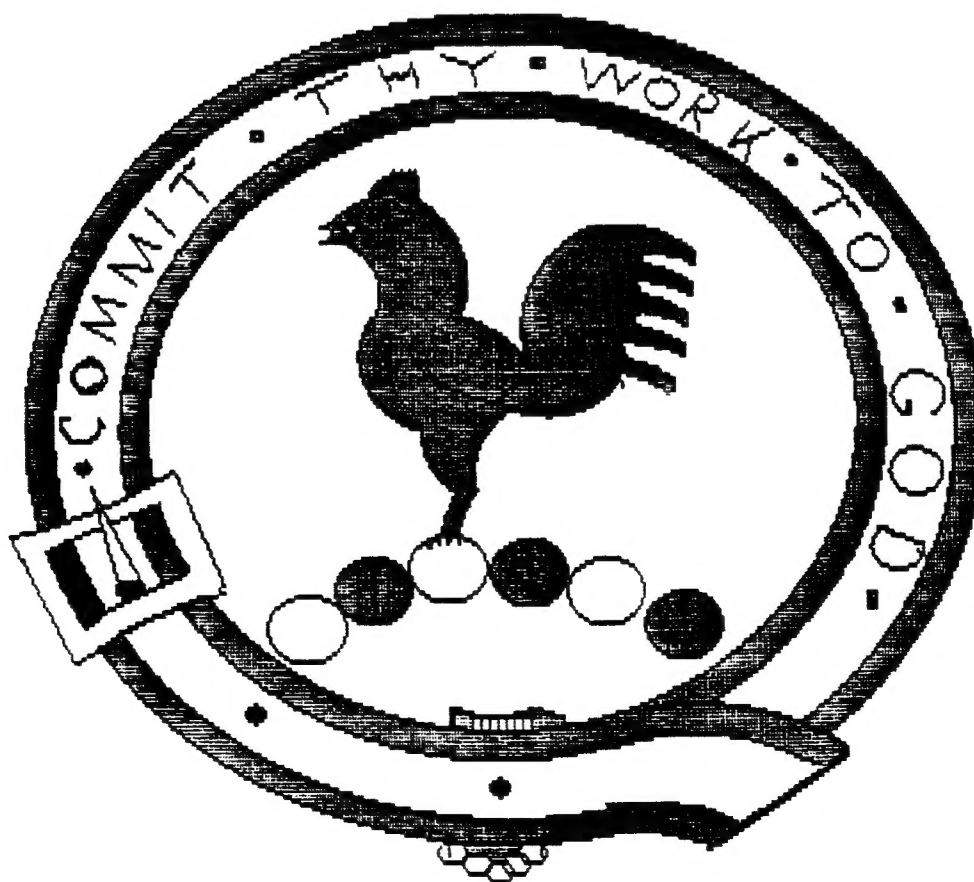
Print neat labels for your floppy disks, showing the filenames present on the disk in columns, in small text, in up to 6 layout styles. You can edit and annotate the text of the label, add headings and notes as required. The layout can be previewed to the screen before printing. Filenames can be sorted into order, or grouped (e.g. Quill "_doc" files together) for a neat, professional appearance. The printer driver is fully reconfigurable for different label sizes and printer control codes. Drivers are supplied for 9 pin and 24 pin dot matrix printers. Suitable for use with our floppy disk labels on printer feed backing, and most disk label sizes. Can print on label rolls up to 4 labels wide. One of our best selling programs, in fact, everyone should have one, and no wonder: "A good, well thought out package, representing excellent value for money" (QL World review, October 1991). Needs at least 128k extra memory. Supplied with printed 20 page manual. Disk Labeller is an ideal companion program for our Disk Indexer program described elsewhere in this catalogue, so that you can both catalogue your disks and tidily label them too. Don't forget about our quantity discounts available for buying more than one program at a time!

SUPER DISK LABELLER (on disk) ... £10.00

Printer roll of 100 disk labels for use with above program £2.50

(no extra postage to pay if ordering at the same time as other items, or add £0.50 postage if just ordering labels)

SINCLAIR CLAN CREST



AN AMERICAN SPECTRUM (THE TIMEX SINCLAIR TS 2068)

The Spectrum was launched here in Britain back in the spring of 1982. It was a massive advance over its' forerunner the ZX81 and, for the first time ever, allowed affordable home computing.

It really was a complete breakthrough. Up until then everyone thought of computers as big mainframe machines with gigantic reel to reel storage, only for use in government departments or large bank head offices. The Spectrum, smaller than a hard backed book and using standard music cassettes really did bring the computer age home to a new generation.

Of course success is nearly always imitated. And the Timex Corporation, who brought modern digital watches to the masses with an affordable price tag decided the time was right to introduce the idea of a small home computer to North America. Out of this desire and a partnership with Sinclair Research here in old blighty a new machine, The Timex Sinclair TS 2068 was born - well almost.

In 1983 the TS 2068 was launched with the familiar Z80A CPU and the same overall operating system as the Spectrum - with equivalent sound and processing abilities to the 48K version. Yet there were differences which made the TS 2068 unique to the US Market.

ADD - ONS

Because of the pace of computer development in the US the TS 2068 has had a very wide range of peripherals and add on features developed for it. Parallels can be drawn between it and the Hobbit - Russia's unofficial Spectrum copy, made as a serious business computer to compete at a much lower price level with PCs and thus equipped with facilities to use PC disk drives, be

linked up to PCs and run complicated Databases and other business software.

Although the TS 2068 was primarily designed

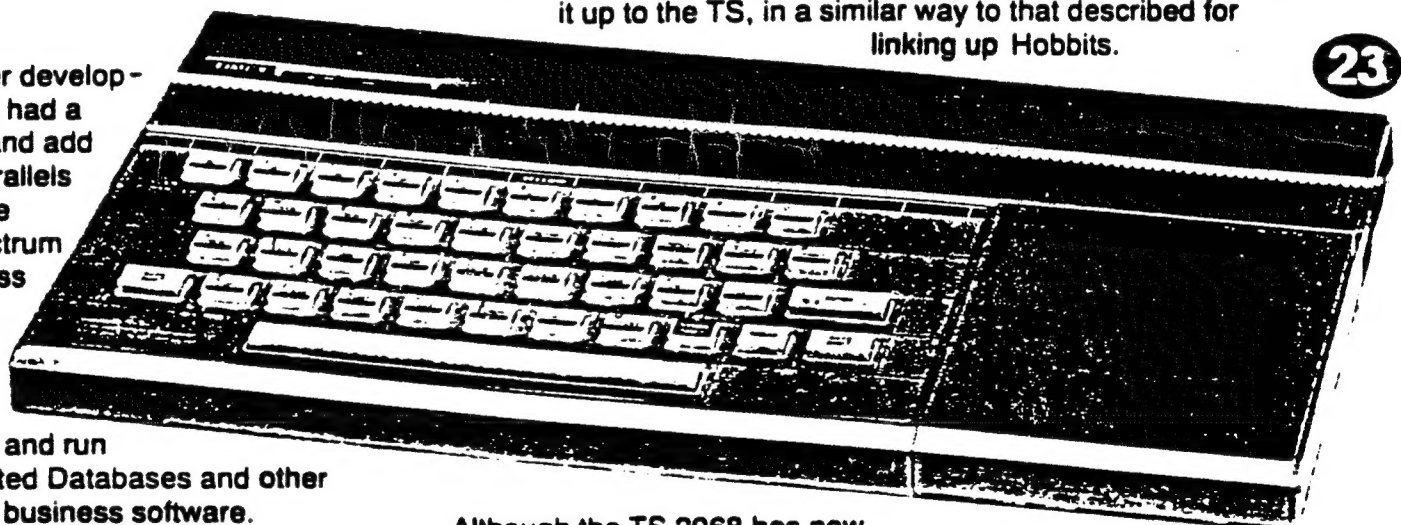
to run using tape loading and saving like our very own Speccy, rather than developing a machine with increased memory capacity like the 128K machine here, American and Canadian boffins simply added on external RAM or linked it up to PC 3.5 or 5.25 disk drives allowing masses of extra memory.

Our contact in America, Bob Shade, who runs a top photography outfit in Philadelphia actually uses his TS 2068s to operate photographic equipment for complicated still work, among other things. Also, because of the problems of Spectrum incompatibility Bob actually runs a Spectrum Emulator on his TS 2068 whenever he wants to run Speccy games or software. He, like many other North American users have brought the machine far beyond the realms of the simple kid's 'home computer' which the Speccy is primarily viewed as here.

Unfortunately for the TS 2068 the pace of PC technology far outstripped it and as these machines became more accessible to the home market as the standard business/home machine (if you use one in the office why not use it at home too) the popularity of the Sinclair went out the window.

However, as well as communicating with similar machines the TS 2068 can also reach out and touch several other computers like the PC and Macintosh using MAClink and PClink. This is done by running programs on the host machine and connecting it up to the TS, in a similar way to that described for linking up Hobbits.

23

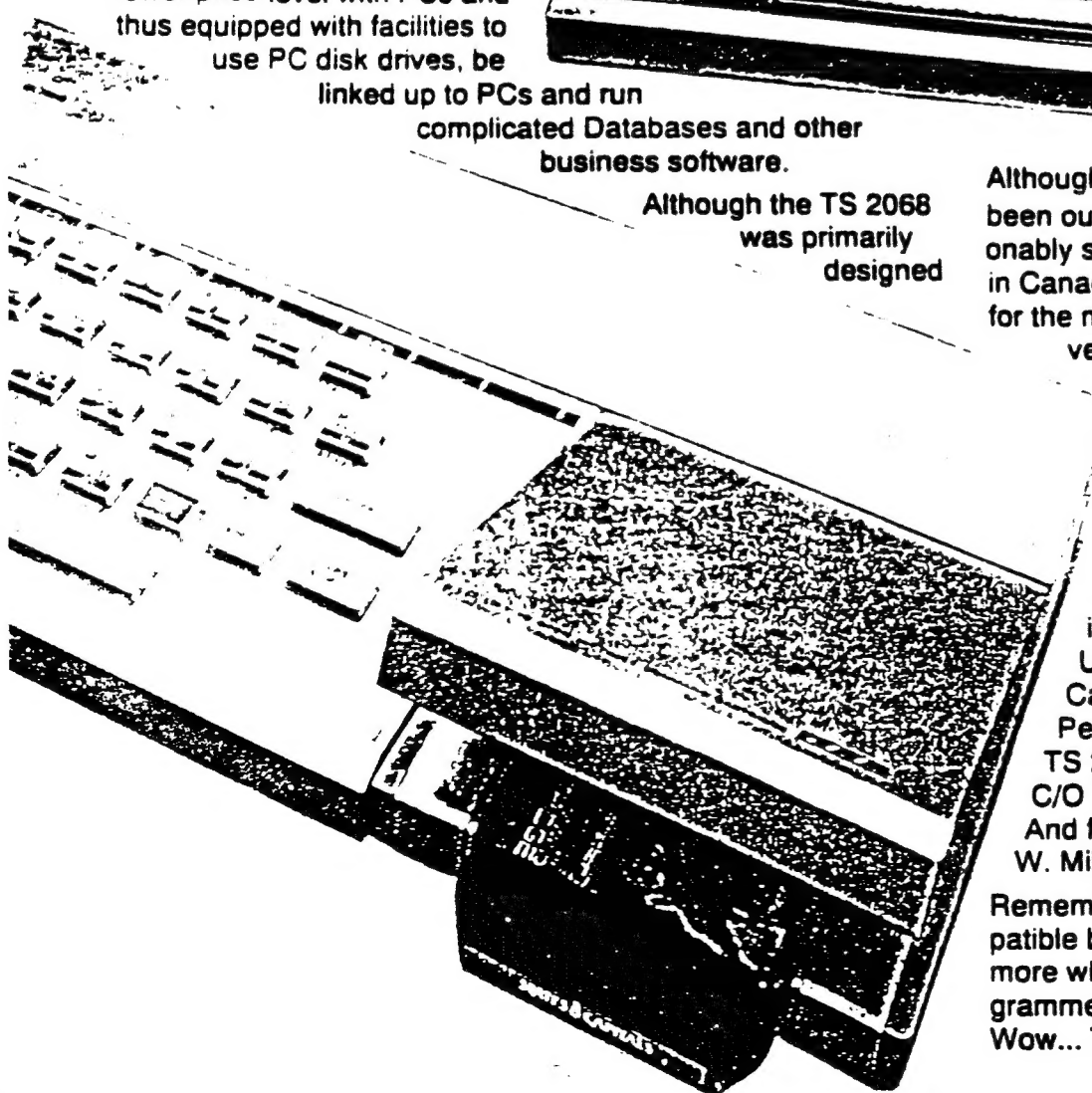


Although the TS 2068 has now been out of production for quite a few years there is still a reasonably strong user group in both the US and more predominantly in Canada. This user group produces most of the new software for the machine and includes Bob Shade himself who has developed a label maker program to make his life easier in the photography business. A Canadian company which supports the TS 2068 with communications devices, memory upgrades etc. is LARKEN ELECTRONICS who have a bewildering array of peripherals for the machine.

USER GROUPS

Addresses to contact user groups for fanzines and information are: Toronto Sinc-Linc, Editor, Timex-Sinclair Users Club, 14 Richome Court, Scarborough, Ontario, Canada M1K 2Y1. Also: Update Magazine PO Box 1095, Peru, Indiana, USA 46970. They deal with Z88 (QL) and TS 2068. Or: Long Island Sinclair Timex Users Group, C/O Harvey Rait, 5 Peri Ln., Valley Stream, NY, USA 11581. And finally SLIX - Sinclair Information Exchange, C/O William W. Miller, 6675 Clifford Drive, Cupertino, CA 95014-4530 USA.

Remember the Timex/Sinclair is not directly Spectrum compatible but there are a lot of similarities. If you want to find out more why not write to one of the user groups. If you're a programmer why not investigate writing something for them? Wow... The Speccy goes international!



TORONTO TIMEX-SINCLAIR USERS CLUB

May 14, 1993

14 Richome Court,
Scarborough, Ont. M1K 2Y1

Les Cottrell
108 River Heights Drive
Cocoa, FL 32922

Dear Les,

Thank you for the \$20 cash for membership renewal. And also thank you for the next few pages of the Larken LKDOS disassembly. We are putting a few pages in each issue of the newsletter. One can have too much of a good thing!! I am still holding your most recent contribution of the disassembly, since Jeff has enough to go on for now.

Two of the items you asked for have been taken. But I am sending the "How to Write your own adventure programs" book. You may find it interesting. But if you are really serious about it, maybe you should ask me for a copy of the Professional Adventure Writer, a program put out for the Spectrum. It has very heavy documentation for it also, maybe 80 pages or so.

Sincerely,

George Chambers

2.5 bid for book
1.35 postage
3.85

